

# BOOKSTORE APP

## *Funcionalidades Requeridas VS Implementação*

### Lista de Livros

**Descrição:** Exibe uma lista de livros sobre desenvolvimento mobile em duas colunas.

**Status:** Implementado numa coluna.

**Observação:** Utilizou-se um JSON local para simular a resposta da API, devido a problemas com as dependências do CURL (detalhado abaixo).

### Filtro de Favoritos

**Descrição** Possibilidade de filtrar a lista para mostrar apenas livros marcados como favoritos.

**Status:** Implementado

### Detalhe do Livro

**Descrição:** Ao clicar em um livro, exibe os detalhes, incluindo Título, Autor, Descrição, e um link para compra que abre o navegador.

**Status:** Implementado

### Adicionar/Remover Favorito

**Descrição:** Permite marcar/desmarcar um livro como favorito, com persistência local.

**Status:** Parcialmente Implementado:

- **Base de Dados Local com SQLite:** Configurada para armazenar livros favoritos.
- **Integração em C++ para Persistência:** Realizada em C++ com interação via JNI para manipular a base de dados.
- **Interface Gráfica de Favoritos:** Incluído um botão de favoritos na Activity de detalhes do livro.

### Integração com a API do Google Books

**Descrição:** A API deve ser chamada via C++, e o resultado retornado por callback.

**Status:** Não implementado devido a problemas de configuração de bibliotecas.

**Observação:** Desenvolveu-se um componente **HttpClient** em C++ para chamadas à **API do Google Books**. Embora a integração completa não tenha sido alcançada

devido a problemas de configuração de bibliotecas (libcurl e wolfSSL), o código encontra-se disponível no repositório e oferece uma base sólida para futuras melhorias.

## ***Desafios***

### **Bibliotecas (libcurl e wolfSSL)**

Houve uma tentativa de integrar o libcurl com wolfSSL para chamadas HTTPS no Android.

**Solução Temporária:** Como alternativa, criou-se um arquivo JSON estático com dados de livros simulando a resposta da API.

### **Persistência de Favoritos**

- a. **Descrição:** A persistência local foi parcialmente implementada para permitir o armazenamento de favoritos entre sessões.
- b. **Status:** Parcialmente funcional. A base de dados SQLite foi configurada em C++ e integrada via JNI, mas a gravação não está completamente operacional.
- c. **Observação Técnica:** Utilizou-se C++ para lidar com a base de dados SQLite e JNI para a integração com a aplicação Android. A funcionalidade de gravação está muito próxima do funcionamento total, sendo uma área a priorizar para melhoria.

## **2. Botão de Filtro**

- a. **Descrição:** A função de filtro está em desenvolvimento e próxima de atingir o comportamento desejado.
- b. **Objetivo Final:** Tornar o filtro intuitivo e visualmente integrado.

## ***Regras de Desenvolvimento e Decisões Técnicas***

**Regras Exigidas:** Toda a integração do back-end foi implementada em C++, e as classes foram importadas para Java através de JNI.

**Integração com a API do Google Books:** Planeada em C++ para realizar chamadas à API, mas encontra-se em fase de ajustes técnicos.

### **Testing**

**Status:** Não implementado

**Observação:** Desafios na implementação completa dos testes devido à configuração complexa com JNI e libcurl.

## ***Próximos Passos e Melhorias***

1. **Correção na Integração com a API:** Revisar a integração de libcurl com wolfSSL para possibilitar chamadas HTTP/HTTPS diretamente na aplicação.

2. **Refinamento do Filtro de Favoritos:** Finalizar a lógica do botão de filtro para uma experiência mais fluida.
3. **Depuração da Persistência de Favoritos:** Concluir a funcionalidade de persistência para garantir que o estado dos favoritos seja mantido entre usos.
4. **Unit Testing**